Section C. Design, Testing and Documentation

Producing an error-free CAI instrument -- Is it possible?

Maureen Kelly, Office for National Statistics (UK)

1. Introduction

The data collected by Social Survey Division (SSD/ONS) are published in high profile Government statistics (for example the Retail Price Index, unemployment statistics). They are widely used by public policy makers. It is therefore essential that the data are as accurate as possible. Computer Assisted Interviewing (CAI) has had a major impact in improving data quality in our surveys. Implementation of a survey instrument is more controlled. For example, interviewer errors in following the correct routing in complex instruments have been eliminated.

These improvements, and the quality of the resulting statistics, depend on the accurate design of the survey instrument. Moreover, CAI has allowed much more complex routing to be designed, since the interviewer is no longer required to work it out on the spot. CAI instruments can include edit checks. As a result, CAI instruments tend to be more complex than paper questionnaires.

Testing plays an essential part in reducing errors in CAI instruments. For large and complex surveys of the kind typical in public sector social research, the theoretical possibilities for testing are almost limitless. The challenge is to find ways of testing both thoroughly and cost-effectively. Surveys that are carried out continuously offer more opportunities to learn from testing than do *ad hoc* surveys. This paper will look at the problems encountered when trying to test the Blaise instruments used for continuous social surveys. The main example will be the U.K. Labour Force Survey (LFS).

An important aspect of producing error-free instruments is the way in which the CAI program is written. This paper will look at ways to structure and write Blaise programs so that errors are less likely to occur, and at ways to provide a basis for efficient testing.

Finally, the paper will look at the different problems encountered when developing and testing completely new instruments as compared with maintaining and amending instruments already in use.

2. Time and resources

The major constraints on the thorough testing of Blaise instruments are the amounts of time and resources that are available for testing. As the theoretical possibilities for testing are almost limitless, if an organisation had infinite resources and no timetables to meet, each instrument could be tested at leisure until it was perfect. In practice, ~~A~~as development timetables are short~~en~~ and survey managers tend to economise on the timetable elements with most elasticity, such as~~to over to~~ testing. However, it could be argued that testing is particularly important~~ when it is probably more essential (as~~ when the instruments

~~may ~~have been written under stringent timetable~~increased~~ pressure that makes them~~and therefore may be more~~ error-prone.~~)~~

Another problem is the increased flexibility that an electronic instrument gives~~n~~ to clients to change the questionnaire specification at late notice~~;~~. It is technically feasible, if not advisable, to make changes right up to~~ usually just before~~ the beginning of fieldwork. The later any changes are made, the less time there is to test them. For example, ~~T~~the LFS was market tested in 1995, after which all aspects of the timetable were accelerated, including the crucial development phase~~;~~ for writing and testing the Blaise instrument. The new contract also allows the clients to make changes to the questionnaire up to two weeks before fieldwork begins. ~~The later any changes are made, the less time we have to test them.~~

Given the constraints on both time and money, whatever testing that is carried out needs to be organised and prioritised in a such a way as to ensure the instrument is as correct as possible whilst making the most efficient use of people's time.

3. The testing process

There are two major strands to efficient testing:

- ensuring that there is a well-structured and systematic procedure for testing the instruments; and
- good management of the testers and the testing process.

~~good organisation of the testing process, that is, deciding who is doing the testing and making the best use of their time; and~~

~~? ensuring the testers are testing the instruments in a well-structured and systematic way.~~

The two strands are inter-related; how the testing can be carried out is, to some extent, dependent on who will be doing the testing.

There are several distinct testing phases that a CAI instrument should pass through before it is used as an interviewing tool. Each phase is important, but~~however~~ on its~~their~~ own it~~ they~~ would not constitute a thorough test of an instrument.

When a new version of an existing instrument is required, an author makes the necessary changes to the CAI programs. The author usually does the first phase of testing; testing the amendments they have made as they go along. This is seldom done ~~is not usually ~~in a ~~very ~~systematic way, as ~~the ~~authors tend~~s~~ to concentrate on the changes they have just made. ~~and~~Authors often fail to~~does not~~ check if there are any adverse~~that the~~ effects of the changes on other ~~have not adversely affected another~~ parts of the program.

Once ~~all ~~the author is satisfied that all the changes have been made, a second stage of testing is required. ~~which~~This is carried out on a completed instrument. It~~This stage~~ is the most important stage and is discussed in further detail below in Section 3.1.

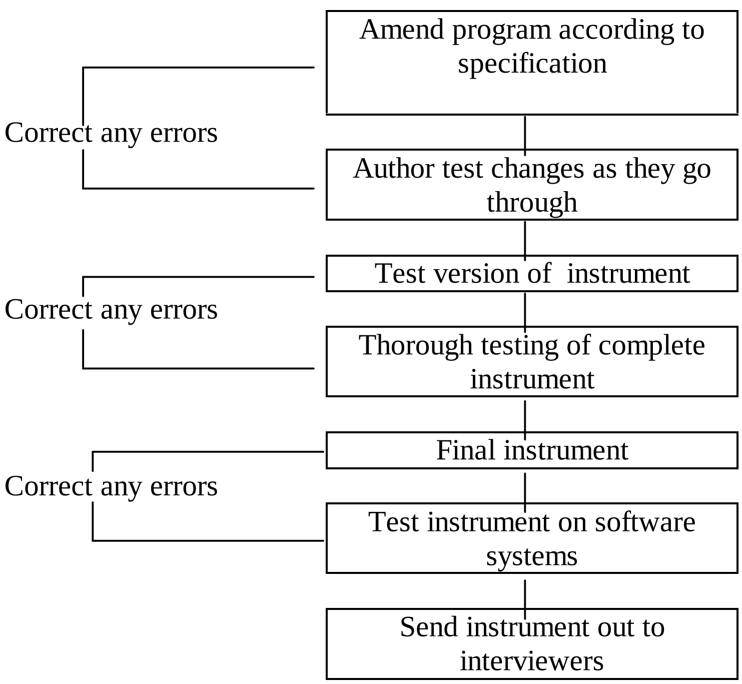
~~All t~~These stages are iterative.~~; i~~If errors are found, the CAI program is amended.~~, t~~The instrument is recompiled and the new instrument is re-tested in the same way. This process is repeated until all the errors have been corrected.

Finally, once ~~all~~ the testing of the instrument is complete, the interface between the final instrument and the software~~operating~~ systems used by the interviewers need to be tested. Parts of the instrument may use~~rely utilise~~ external information or programs (for example, DOS environment variables, Dynamic Link Libraries and external Blaise files and libraries). In addition, the interviewer's systems may need to use information written out in ASCII from the Blaise instrument. (The need for this is reduced in organisations that use Maniplus for case management.)~~, writing out of ASCII files).~~ For the LFS, there are two interviewers' systems with interfaces to Blaise:

- Casebook, used by the face-to-face interviewers on their laptops, and
- a CATI~~PC-based~~ Call Scheduling system for the telephone interviewers.

Once the testing is complete, the instrument can be made available to the interviewers, to conduct interviews. Figure 1 shows a summary of the testing process.

Figure 1



3.1 Structured testing of the complete instrument

Thorough testing requires independent testers. There are two common methods of testing instruments: entering data into the instrument; or examining the source code. Only someone who can read CAI programs can do the latter. In~~Within~~ SSD/ONS, many of the independent testers are unfamiliar with Blaise programs. Therefore most~~the majority of the~~ testing is done interactively by inputting data into the instrument. Examination of the source code is usually only used when to trying to correct a ~~identified~~ problem that has been identified by the first method~~with the code~~. Moreover, the only way to test layouts, question wo~~o~~rding and text substitution is by looking at the instruments on screen.

An independent specification, that is, one that has not been generated by the CAI program itself, is a necessary tool for testing. Firstly, the author~~s~~ needs it to make all the necessary changes. Secondly the independent testers should use it to check that~~ese~~ the changes have implemented correctly. The~~is~~ specification needs to be understandable to testers unfamiliar with the CAI programs. For this reason, many survey researchers use flow charts to show the routing of the instrument. Survey researchers do not always have formal specifications from customers. Customers sometimes state their requirements for questions in general terms. The researcher then has to translate these requirements into a specification and routing for the CAI program.

Tests of~~When testing~~ routing, checks and signals should examine all possible combinations of data ~~should be tested~~. It is important to check ~~it is important~~ that the ‘negatives’ are tested as well as the ‘positives’. That is, that the correct sub-groups of people are asked the question and that people not in the specified group are not asked the question.~~,~~ It is, of course, far worse to collect too little information than too much. If too much is collected, the data can be edited after the interview. However, asking respondents questions which do not apply to them is to be avoided if possible, to reduce respondent burden and costs, and avoid disruption of the flow of the interview ~~would be affected as respondents are asked questions that are not applicable to them~~.

The ‘negative’ testing is always more time consuming then testing positives. There are usually more combinations of routing that result in the question not being asked than being asked.

Example 1

Five questions QA, QB, QC, QD and QE have the answer categories ‘yes’ and ‘no’ and the routing is as follows:

```
QA
IF (QA = YES) THEN
   QB
    IF (QB = YES) THEN
          QC
    ENDIF
ENDIF
QD
IF (QA = YES AND QD = YES) THEN
   QE
ENDIF
```

To test QB, four responses to QA need to be tested: yes, no, don’t know (Ctrl+K) and Refusal (Ctrl+R). These give the following results:

QA	***QB***
Yes	Asked
No	Not Asked
Don’t Know (Ctrl+K)	Not Asked
Refusal (Ctrl+R)	Not Asked

It is important to test that the correct question attributes have been used and what effect the missing values (Ctrl+R, Ctrl+K) have on the routing of the instrument. They are sometimes overlooked, but they can have an affect on the routing. In this example, it is fairly obvious that the Don’t Know and Refusal will not be routed to QB. In others it is not so clear.

As the routing gets more complex, the number of different combinations that need to be tested increases. The routing to~~of~~ QE involves two variables: QA and QD. Each question has four possible answers, resulting in 16 combinations. Only one of them should result~~ing~~ in the question being asked:

QA	QD	QE
Yes	Yes	Asked
Yes	No	Not Asked
Yes	Ctrl+K	Not Asked
Yes	Ctrl+R	Not Asked
No	Yes	Not Asked
No	No	Not Asked
No	Ctrl+K	Not Asked
No	Ctrl+R	Not Asked
Ctrl+K	Yes	Not Asked
Ctrl+K	No	Not Asked
Ctrl+K	Ctrl+K	Not Asked
Ctrl+K	Ctrl+R	Not Asked
Ctrl+R	Yes	Not Asked
Ctrl+R	No	Not Asked
Ctrl+R	Ctrl+K	Not Asked
Ctrl+R	Ctrl+R	Not Asked

12 out of the 16 combinations involve the Ctrl+K and Ctrl+R categories. In many cases the Don't Knows and Refusals are treated in the same way. To save time, ~~test~~ in most cases it would be sufficient to test either Ctrl+K or Ctrl+R at each question but not both. In the example given above this reduces the number of combination from sixteen to nine. Note that~~If a question has different attributes, for example~~ when there is an explicit 'don't~~k~~ know' response category ~~and therefore~~ the attribute NODK is often attached to avoid giving the interviewer two 'don't know' options. A question like this should be checked to make sure Ctrl+K cannot be entered. If the question is used in the routing for other questions the routing of Ctrl+R has to be tested.

Particular attention should be paid when NOT conditions are used in routing. It is may be unclear what should be routed to the question, and particularly easy to overlook the routing for the missing values.

Once the routing for QB has been satisfactorily tested, then QC can be tested, and so on. The nesting of the questions within a block facilitates this testing process. If one question relies totally on the information from a preceding question that has already been tested, then you do not need to go back and retest the routing to the preceding question. For example, the routing of QC relies totally on QB. Once the routing to QB has been tested and found to be correct, QC can be tested without the need to go back and retest the routing to QB.

In an ideal world the routing to all the questions within the instruments would be tested. This would ~~usually~~ take more time than is ~~usually~~ available for testing. The testing can be separated into several stages so that the instrume~~em~~nt is thoroughly tested without having to test every single question:

- testing the changes made;
- testing the remainder of the instrument;
- overall testing of the instrument.

~~? testing late changes.~~

~~T~~3.1.1 Testing changes

The main focus of testing amended instruments for continuous surveys is to ensure that all the necessary changes have been implemented correctly. There are different types of~~different~~ changes that can be made to an instrument, all of which need to be tested–:

- new questions introduced;
- old questions removed;
- changes to routing;
- changes to wording – question text and answer categories;
- changes to answer ranges;
- new/amended checks and signals;
- new/amended text substitution;
- changes to computations;
- changes to layout;
- changes to attributes (e.g. EMPTY, DON'T KNOW, REFUSAL)
- changes to parameters;
- updating of coding files.

In some instances, the changes may fall into more than one category. For example, ~~. W~~when adding a new question, all the wording, routing and layout need to be checked. Changing the routing to an existing question may also result in checks, signals and~~,~~ text substitution being amended. Adding an extra answer category to a question may also require a layout change to ensure that it fits on the same screen as the original answer categories.

Routing, checks, signals, computations and text substitution can be checked at the same time together as they all appear and are affected by the RULES paragraph of the program. Similarly~~In a similar way~~, question wording, layout of the question text and on-screen instructions can be checked together.

When testing changes, a 'test' version of the instrument can be used. This shows fields and auxfields that would normally be hidden in order that computations can be easily tested. The author has to remember to hide these fields before the instrument is finalised. It is usually a good idea to make testers aware that some fields appear which they will not see in the final version.

3.1.2 Testing the remainder of the instrument

Once all the code for the intentional changes to the instrument has~~ve~~ been tested, the rema~~am~~inder of the instrument needs to be tested to ensure that none of the new code has~~changes have~~ adversely affected the conditions for~~routing to~~ any other questions~~variables~~, checks or computations. The hierarchical block structure of the CAI program and the nesting of questions within these blocks facilitate this process.

If the routing to any questions, ~~or~~ checks~~,~~ or computations~~,~~ remains as before~~have not changed~~ but the conditions for these items~~y use~~ include variables for which~~where~~ the~~y~~ routing has been changed, then these items too should be tested. ~~This includes both questions within the same block as the amended question, and in other blocks.{tm - I don't understand the need for the last sentence}~~

If the routing to a particular question is amended, the routing to all the subsequent questions within the same block should be tested. Simple mistakes such as putting an ENDIF in the wrong place can cause ramifications throughout the routing of the remainder of the program.

Example 2

The specification for four questions is as follows and an amendment needs to be made to QB:

QA – ask everyone

QB – ask only if QA = YES (previous routing – ask everyone)
QC – ask everyone
QD – ask only if QC = YES

The routing was written as:

IF QA = YES THEN

```
  QB
  QC
ENDIF
IF QC = YES THEN
   QD
ENDIF
```

The routing to QB (the IF condition) is correct. The ENDIF that should come after QB was placed incorrectly after QC. Therefore the routing to QC is incorrect. Even though the routing to QD is written correctly, not all the people who should be asked QD are routed to it. The error in the routing to QC is carried through to QD. After testing the change to QB, there still needs to be a test of routing to QC and QD if the tester is to find the error.

~~If a block has not been amended~~Sometimes a whole block is unaffected by the current revision. No code within it needs to be changed. None of the conditions for any of its component questions invokes amended questions in other blocks. The routing to unamended blocks still need to be tested. In the above example, the questions QC and QD could relate to blocks of questions. The error would then affect two whole blocks of questions.

To test the routing to a block, some questions within that block need to be tested. Only certain questions need to be tested: the first question in the block, any other questions at the highest level within the block (that is, with no IF/ENDIF statements), and questions with references to questions outside the block. ~~-~~The nesting of questions within a block m~~n~~eans that most questions are dependent only on answers to previous questions within that block. ~~, and t~~O~~herefore o~~nly a few questions ~~that~~ have references to questions outside that block. ~~Testing the routing to all the 'un-nested' questions is sufficient to test that the whole block is correct. 'Un-nested' questions are the first question in the block, those at the highest level within the block (with no IF/ENDIF statements), and those with references to questions outside the block. The routing within the block can be assumed to be correct before any amendments were made to the program. The~~

~~CAI programs used before the amendments are usually the ones used to create the instruments that are currently in the field.~~ Testing the ~~'un-nested'~~first and highest level questions ensures that the routing to the block is correct. Testing questions with references to questions outside the block ensures ~~and~~ that the transfer of information between blocks is correct. Testing the routing to all the 'un-nested' questions is sufficient to test that the whole block is correct. This process should be repeated for each block within the program, so the whole instrument is tested.

3.1.3 Overall testing of the instrument

Once the testers are satisfied with the changes to ~~all the necessary~~ individual questions~~ have been satisfactorily tested~~, they should carry out more general testing of the instrument ~~questionnaireis should be carried out~~. Many surveys, including the LFS, do this by entering ~~On the LFS, they Imaginary households are created and enter~~ data ~~enter~~ for imaginary test households~~ed into the instrument~~ to check that the questions appear as expected. The LFS questionnaire is made up of a household section followed by~~be and~~ individual~~a series of~~ interviews for each person in the household. Imaginary ~~Several~~respondents are created to reflect the important sub-groups the LFS is trying to identify, such as~~. For example~~: full time employees; full-time students with and without a part-time job; housewives, not working and looking after small children; retired old age pensioners; and unemployed people looking for work.

The~~is~~ next phase of ~~is followed by~~ testing is for different types of households and the full range of~~testing the various~~ household outcomes. As with testing 'negative' routing, it is important to check~~that~~ the Blaise code applying to ineligible and non-responding households~~ are checked and are working correctly~~. Most of the testing concentrates on what occurs during an interview with ~~, and that~~ respondents~~ are asked the correct questions~~. Experience from the LFS has shown that interviewers can encounter serious problems, such as unresolvable check messages, if testers overlook cases where there is no respondent.~~this part of the testing is overlooked.~~

Similar tests are appropriate for surveys of persons rather than households.

~~{Tm - I'm not sure that this section adds anythingLate changes~~

~~There are usually late changes made to the program just before it is finalised when there is no time to fully test the instrument again. Depending on what these changes are, it is may be advisable, but not essential, for another person to check the amendments. For example, if a spelling mistake is corrected it is probably not necessary to check it. However, if an important computation or piece of routing is changed, someone should check that the amendments are correct.}~~

3.2 Organisation of the testing process

3.2.1~~a)~~ Who should do the testing

In~~Within~~ SSD/ONS, the author of the Blaise instrument ~~usually~~ has overall responsibility for ensur~~test~~ing that the instrument ~~to ensure that it~~ delivers what is expected. Although the author is usually involved in the testing it is not usual, or advisable, that they alone should~~author solely~~ test~~ed~~ the questionnaire~~ alone~~.

Authors can sometimes let errors slip through as they can suffer from 'seeing what is expected' rather than what is there. Therefore other people should be involved in the testing: other researchers working on the project, administration or field staff, and interviewers. The number of people and amount of time available for testing will depend upon the project and the resources available. On the LFS, at least one other researcher and ~~alongside~~ a number of ~~telephone~~ interviewers from the CATI unit test the instrument. Since the unit is centralised, it is easy to discuss the instrument with these experienced interviewers and costs are low. Testing by interviewers, in addition to other testing, is the norm for CAI instruments in SSD/ONS.

Provided there is careful management,~~Usually~~ the more people who look at the questionnaire the better. For example if there is only one day set aside for testing, it would probably be better to use two people to look at it for half a day each, rather than one person for a full day. This has the advantage that one person may pick up~~you~~ an error the other has ~~accidentally~~ missed. ~~Also, t~~Testing questionnaires is a laborious and dull task, so the time any one ~~and~~ tester spends on the task should be short.~~s are likely to get bored, and therefore may be more likely to miss errors if they have to test the instrument for a long period of time. {Tm - I'm not clear from this para and the next if we want to check each other's work or not - the paras seem to give conflicting advice. I think one of the paras has to go.}~~

Only a moderate amount of duplication of testing amongst testers is necessary. Poorly organised testing leads to excessive duplication can occur which is a waste of valuable time and resources. ~~The more people involved in the testing, the more likely there is to be a unnecessary amount of duplication of effort, with the testers testing the same part of the questionnaire at the same time and therefore finding the same errors. Although a moderate amount of duplication is inevitable and necessary, so that more than one person has tested each part of the questionnaire, too much is just a waste of time and resources.~~

~~In the past, w~~When ~~the~~ LFS interviewers first became involved in testinged the questionnaire, they were given ~~very~~ little guidance or co-ordination ~~as to how they should test the questionnaire~~. Unsurprisingly, they all tended to start ~~This resulted in the interviewers usually starting to test the questionnaire~~ at the beginning and duplicate their efforts and findings. ~~, and all coming across the same errors. It also meant that~~ M~~m~~ore time was spent on~~testing~~ the beginning of the questionnaire than on its later sections~~, finding the errors which left less time for testing the latter parts of the questionnaire~~.

One way the LFS sought to reduce the amount of duplication was to split the testing of the changes into sections. The testers ~~esters~~ are made responsible for testing individual sections, such as household information, employment details, job search activities, education, administration etc. The work is organised so that each section is tested by at least two people. Splitting the questionnaire into sections also made the task more manageable for the testers, some of whom admitted that they felt daunted by having to test the entire questionnaire. The~~is~~ results of giving the testers ownership of a clearly defined and manageable task is that ~~in~~ each section of the questionnaire is~~, being~~ thoroughly tested. In the previous situation, a~~rather than~~ number of people ~~superficially~~ checked~~ing~~ the whole questionnai~~ia~~re rather superficially.

3.2.2~~b)~~ Documentation

Testers need documentation so that they know what to test. As well as the independent questionnaire specification, the LFS testers also found the following documents useful:

- T~~T~~he specification for the previous version of the instrument. As they were testing the amendments to an existing questionnaire, they found it useful to be able to compare the two specifications, as well as the instruments.
- A list of all the changes made to the instrument, which they would have to test.

- A list of all the other questions that needed to be tested (see 3.1.2 – Structured testing. Testing the remainder of the instrument).
- A structure diagram of the instrument. Many of the testers are unfamiliar with Blaise and the block structure of the instrument. They found a structure diagram useful to see how all the sections fit together, which is not always obvious from a paper specification.
- A set of test data ~~list~~ of imaginary respondents and households to test the whole instrument.

3.2.3~~c)~~ Communication

Good communication is essential for an efficient testing process. This includes communication amongst the testers and also between the testers and the program author. One problem identified by ~~with~~ the LFS CATI interviewers in their roles as testers is that they work fairly independently of each other. It is particularly difficult to share information across shifts. ~~, working on different shifts in the Telephone Unit.~~ One tester could discover a problem and inform the author, only to discover~~not realising~~ that another tester had already found it. Email was ~~usually~~ used to communicate between the author and the testers~~,~~. This ~~which~~ provided a useful record of the reported problems for the author, but not for the testers. To improve communication, a~~However, the testers also decided that a~~ log of the problems found was set up.~~would also be useful, which~~ It listed: a description of the problem, which variables it affected; which tester had found the problem; when it was found and when it was corrected.
The log was particularly important in recording when a problem had finally been corrected. ~~It was important to find out when the problem had been corrected as testers could spend a lot of time checking known errors before they have been corrected.~~ The testers usually get more than one version of the questionnaire to test: they test the original version and~~, they test it,~~ find some errors;~~,~~ the author corrects the~~m~~ errors and produces~~gives them~~ another version. The~~y~~ interviewers then test th~~e new version~~~~is~~ to see if the errors have been corrected. This process is continued until all the errors have been eliminat~~correct~~ed. However, there are some problems which are harder to solve than others, and it may be the case that the author has to ~~you~~ issue an interim version of the instrument, which only corrects some but not all of the reported problems. It is essential that the author communicates to the testers when the amended instruments are available to use and which problems have been corrected, otherwise vital testing time can be wasted.

3.2.4 Timetable

One of the major problems encountered when testing instruments is the time available for testing. It is important for all the people involved in the process to know when~~whne~~ the testing will take place, and how long they will have to test the instruments. This includes when they will get revised questionnaires. This will help the testers plan their work and make the most of the time available.

4. Good practice in questionnaire design

So far, this paper has focussed on the issues surrounding the problems of testing instruments after~~, once~~ they have been changed. Another important aspect of providing error-free instruments is trying to avoid producing the errors in the first place. There are many ways to improve instrument design. Two common examples may illustrate the point.

The simplest way to minimise error is to make as few changes as possible to a well-tested instrument. ~~Usually, the less changes that are made to a program the smaller the number of errors can occur. However, when testing instruments for continuous surveys the majority of the testing concentrates on ensuring the~~

changes have been implemented correctly. Even in these cases, the way in which the program is written can have an affect on the number of errors that are likely to occur. Before the LFS moved to Blaise III (in September 1996), there was a major revision of the program each quarter. Every year, About 10% of the instrument changed the similar changes were made to each quarter. The quarterly instruments were each as large as Blaise 2 could handle's questionnaire. When the LFS moved to Blaise III made itit was possible to create an annual questionnaire, which contained the routing for all four quarterly questionnaires. This consolidatedreduced the amendments made to the questionnaire into a single annual revision and thus reduced the potential for errors. On other surveys, Blaise III made it possible to consolidate previously separate household and individual instruments, resulting in similar improvements Now, only one major revision of the questionnaire is carried out each year.

Good design of routing is critical in reducing the scope for errors. Routing should be at the highest level possible.; Iif a particular condition applies to all the questions within a block, that condition should be used in the routing to the block rather than the routing to the questions within the block. This will reduce the number of questions that would need to be tested after all the intentional changes within a block have been tested (see 3.1.2 – Structured testing. Testing the remainder of the instrumentpart (b)). Also, dDuplication of routing at the two levels, which sometimes occurs under time pressures if the Blaise writer does not plan changes carefully, -is particularly toshould be avoided. Not only is it inefficient,, butas it is easier for errors to creep in later when one level is amended and the other is not.

It may be useful to create dDerived variables (DVs) can be used for complicated conditions that are used in several different places within the program and are likely to change. For example, on the LFS the questions on on work-based government schemes regularly change. They are widely used in the instrument. The current definitionand is currently defined as (YTETMP =1,2,4) OR (SCHEME = 5) OR (PROJWK = 2, 4). AIf a derived variable (DV) might usefully bewas computed for this set of conditions and then used throughout the instrumentprogram., Wwhen the definition changes,d only the DV would require amendment. At present, if the definition changes, the routing to all individual questions which use that definition in their conditions requires amendment. Errors are therefore more likely to occur than if a DV is used. The fewer changes that need to be made, the less likely errors are to occur. It should be noted that good practice requires the name of the DV to be changed if its substantive content changes. If this is not done, there is a risk of introducing errors at the analysis stage. Analysts of the data may assume - reasonably but, in this case, incorrectly - that a variable with a single name has the same meaning throughout a longitudinal dataset. It is easy to add an incremental number to the DV name through the search and replace function.

5. Differences between developing new instruments and maintaining instruments for continuous surveys

This paper has concentrated on the amending and testing of instruments for continuous surveys. This accounts for a considerablemajor part of the work carried out within SSD/ONS. However, SSD/ONS also does manya number of one off *ad_-hoc* projects using CAI. The instruments for these surveys are completely new. They need to be designed and tested. As with continuous surveys, *ad_-hoc* surveys must be designed under stringentalso suffer from the budgetary and time constraints. However tThe major difference between instruements for *ad_-hoc* and continuous surveys is that testers cannot concentrate on particular parts oftesting changes to the instrument, as everything is new.

One way in which SSD/ONS has tried to address this problem is to use standardised blocks and templates. For example, there are standard blocks to ask questions on economic activity, and household composition. The author does not then have to worry about writing or testing these blocks and can concentrate on the survey--specific blocks. The routing within these standard blocks has already been tested. When the testing

~~the~~ *ad hoc* instruments, the testers can test the routing to these blocks ~~is correct~~ in the same way as they would test unamended blocks on a continuous survey.

6. Conclusion

Producing error-free instruments is vital in producing good quality data. Practical survey work is done under stringent limits on time and resources, so it is impossible to test literally every combination of the variables and their categories. Fortunately, the strong structure-building features of Blaise and the possibilities for good design provide a basis for prioritising the testing. ~~It is extremely difficult, if not impossible, to create a perfect Blaise instrument, particularly given the time and resources available to survey producers.~~ Nearly a decade of continuous and ad hoc survey work with Blaise in SSD/ONS has shown that data quality is much improved over paper questionnaires in the areas of instrument design that we attempt to improve through testing, such as routing. Testing aspires to be as scientific as possible, but the constraints under which it is done mean that there is value in the kinds of practical experience recorded in this paper.

.

See also 'Blaise Testing Protocol' for a fuller list of all items that can be tested. Blaise Services at Westat, January 1997